Description

COUNTERING SPAM THAT USES DISGUISED CHARACTERS

Inventor: Shaun P. Cooley

Technical Field

This invention pertains to the field of countering spam that infects electronic messages by disguising characters.

Background Art

As used throughout this specification including claims, "spam" is any electronic message that is unwanted by the recipient; and a "clean" electronic message is one that is not spam. The amount of spam sent over computer networks has increased with the increasing popularity of electronic messaging schemes such as e-mail. Spam filters have been designed to counter the flood of spam. However, spammers have employed various tricks to neutralize the spam filters and thereby place their unwanted messages in front of recipients.

Once such trick employed by spammers (illustrated in Figure 1) is to break up the electronic message 1 into two portions: a visible portion 2 that is visible to the human recipient and readable by the spam filter, and an invisible portion 3 that is invisible to the human recipient but nonetheless readable by the spam filter. The visible portion 2 contains the spam message, typically between 10 and 20 words long, while the invisible portion 3 is much longer, typically between 1000 and 2000 words long. The invisible portion 3 contains characters that lull the spam filter into concluding that the message 1 is clean. In the case where the spam filter is a statistical filter (such as a Bayesian filter, a neural network, or a support vector machine), the invisible portion 3 of the message contains many more words than the visible portion 2. Furthermore, the invisible text 3 contains words that are innocuous. Since the spam filter processes many more innocuous words from the invisible portion 3 than spam words from the visible portion 2, the spam filter erroneously concludes that, as a whole, the message 1 is clean.

This spamming technique can be used with any spam filter that takes into account characters within the message 1. In the example shown in Figure 1, if the spam filter has been programmed to conclude that a message 1 is clean when the word "cancer" appears in the message 1, the spammer can place the word "cancer" in the invisible portion 3 of the message, counteracting the effect of the word "breast" in the visible portion 2 of the message. (The word "breast" would normally trigger the spam filter to conclude that the message 1 contains spam.)

The present invention provides methods, apparati, and computer readable media to counter the above-described spamming technique.

Disclosure of Invention

Computer-implemented methods, apparati, and computer-readable media for countering spam that disguises characters within an electronic message (1). A method embodiment of the present invention comprises locating (36) portions of the electronic message (1) where the difference between foreground color and background color is negligible; deleting (37) from the electronic message (1) foreground characters from said portions, to form a redacted electronic message; and forwarding (33) the redacted electronic message to a spam filter (23).

Brief Description of the Drawings

These and other more detailed and specific objects and features of the present invention are more fully disclosed in the following specification, reference being had to the accompanying drawings, in which:

Figure 1 illustrates an electronic message 1 that has been composed using a spamming technique of the existing art that is countered by the present invention.

Figure 2 is a diagram illustrating apparatus usable in carrying out the present invention.

Figure 3 is a flow diagram illustrating a method embodiment of the present invention.

Detailed Description of the Preferred Embodiments

As used throughout this specification including claims, the following terms have the following meaning:

"HTML" is HyperText Markup Language, a common language used by the World Wide Web sector of the Internet.

"Electronic message" 1 is any message that is in electronic or digital form. Thus, for example, electronic message 1 can be e-mail, an instant message, a chat room message, a newsgroup message such as an Internet newsgroup message, a wireless message such as Morse code modulated onto an electromagnetic RF carrier, an SMS (Simple Messaging Service) message, an MMS (Multimedia Messaging Service) message, an EMS (Enhanced Messaging Service) message, or a text or graphics pager message.

"Rendering" means converting an encoded message into human readable images and/or text that can be portrayed on a display device. In HTML, an image is rendered pursuant to an IMAGE tag.

"Character" is any computer-representable mark, such as an alphanumeric character, a special symbol like = - % or $, a peace symbol, a design, a trademark, a cartoon, graphics, etc. A character can be from any natural language.

"Natural language" is a language that is spoken and/or written by humans.

"Word" is a group of characters.

"Coupled" encompasses any type of coupling or connection, whether direct or indirect.

With reference to Figure 2, "user" refers to a computing device 5 and/or a human who has control of computing device 5. Device 5 is broadly defined herein as any type of computer or any type of device containing a computer. Thus, device 5 may be an individual client computer such as a personal computer (PC), laptop computer, handheld computer, etc.; an enterprise computer such as a workstation, gateway computer, or proxy computer; a two-way pager; or a messaging telephone.

User 5 sends and receives electronic messages 1 to and from a network 4. The network 4 may be any type of wired or wireless network, such as the Internet, the public switched telephone network (PSTN), a local area network (LAN), or a wide area network (WAN).

There can be a plurality N of user devices 5. They may be associated with some enterprise, e.g., a corporation, a university, a set of affiliated users 5 connected to each other by a local area network, etc.

"Foreground" of an electronic message 1 is the region or regions of the message 1 where information consisting of one or more characters is conveyed to the recipient user 5.

"Background" of an electronic message 1 is the region or regions of the message 1 other than foreground.

Spammers can make foreground characters invisible by changing the color of the foreground characters to match the color of the background or, conversely, by changing the color of the background to match the color of the foreground characters.

"Color" is a quality of visible phenomena having hue, saturation, and brightness.

"Hue" is that attribute of color in respect to which the color may be described as red, yellow, green, blue, or intermediates thereof. Hue is expressed in degrees from 0 to 359. 360 degrees of hue equals 0 degrees of hue.

"Saturation" is that attribute of color in which the color may be differentiated from another color as being higher or lower in degree of vividness of hue; that is, as differing in degree from gray. Saturation is expressed in percent, from 0% to 100%.

"Brightness" is that attribute of color which measures its position on the white to black scale. Thus, a dark gray has a low brightness, a medium gray has a medium brightness, and a light gray has a high brightness. Brightness is expressed in percent, from 0% to 100%.

"Gray-scale color" is a color having a saturation of zero percent.

"Hued color" is a color other than a gray-scale color.

A color is either a gray-scale color or a hued color.

To implement the present invention, a given user (arbitrarily illustrated as user 5(1) in Figure 2) has associated therewith a parser 21, an optional color comparison module 22, and a spam filter (spam detection engine) 23. Parser 21 is a module that performs semantic analysis on messages 1. In the case where message 1 is e-mail, parser 21 is a HTML parser. Parser 21 is usually part of a renderer. Parser 21 has the capability of converting text (which might be in ASCII format) into a format more suitable for subsequent programming, e.g., binary. Parser 21 may comprise or be coupled to ancillary components such as a processing unit, comparison module, etc. These ancillary components are useful in assisting parser 21 to perform its duties as broadly described herein.

Coupled to parser 21 is optional color comparison module 22. The purpose of module 22 is to determine, for non-simple cases, which portions, if any, of message 1 are invisible or nearly invisible to a typical human user 5. Any such portions 3 are deleted by parser 21 before parser 21 sends the message 1 to spam filter 23.

Spam filter 23 is coupled to parser 21 and can be any type of spam filter that is influenced by characters within message 1, such as a machine learning based spam filter, a neural network, a Bayesian classifier, a support vector machine, a non-machine learning based spam filter, a fuzzy hash filter, a collaborative filter, an RBL filter, a white list/black list filter, etc.

Optional stack 25 and optional flag 26 are coupled to parser 21. Stack 25 and flag 26 each consist of any type of storage means, such as a register, RAM memory, state of a state machine, area on a hard drive, etc.

Modules 21, 22, 23, 25, and 26 can be implemented in software, firmware, hardware, or any combination thereof. When implemented in software, all or portions of said modules 21, 22, 23, 25, and 26 can reside on a computer-readable medium such as a hard disk, floppy disk, DVD, CD, etc, or on a plurality of such computer-readable media.

The operation of the present invention will now be illustrated in conjunction with Figure 3. The method begins at step 31. At step 32, parser 21 asks whether any portions of message 1 remain to be processed. If there any no such portions left to be processed, parser 21 (at step 33) sends message 1 to spam filter 23, where filter 23 processes message 1 in a manner that is normal and customary for filter 23.

If there are portions of message 1 remaining to be processed, the method proceeds to step 34, where parser 21 examines the next color tag within message 1. A color tag is any means by which the sender of message 1 has indicated a color in which a portion of message 1 will be rendered on a display associated with recipient computer 5. In HTML, there are several ways of providing color tags, including inline style, color attributes, background attributes, and style sheets. These are illustrated below:

Inline style:

```
<P style="color: white; background-color: black">This text is visible</P>
```

Color/background attributes:

```
<P><font color="white" background="black">This text is also visible</font></P>
```

Style sheets:

```
<STYLE>
.WhiteOnBlack { color: white; background-color: black}
.WhiteOnWhite { color: white; background-color: white}
</STYLE>
<P class="WhiteOnBlack">This text is visible</P>
<P class="WhiteOnWhite">This text NOT visible</P>
```

In the above example, color attributes have been combined with background attributes, but they could be separated from each other. Note that in each of the above examples, a color tag is preceded by a “less than” sign.

At step 35, parser 21 determines whether the present color tag being examined indicates that the color of either the foreground or the background has been changed by the present color tag. If not, the method reverts to step 32. If the color has changed, however, the method proceeds to step 36, where parser 21 determines whether the difference between the new foreground color and the new background color is negligible. This step 36 may or may not require the assistance of color comparison module 22. If the difference between the foreground and background colors is negligible (i.e., zero or very small), this indicates that the foreground is invisible or nearly invisible to the typical human user 5. Therefore, this portion of the message 1 is deleted by parser 21 at step 37, and the method reverts to step 32. At least the foreground characters from said portion are deleted; possible the entire portion, including background, is deleted.

If, however, the result of the analysis at step 36 indicates that the difference between the foreground and background colors is not negligible (i.e., the difference is greater than a small amount), this is the equivalent of saying that the foreground is visible to a typical human user 5, and therefore foreground characters from this portion are left in the message 1 by parser 21 at step 38. After execution of step 38, the method again reverts to step 32.

It can be seen from the above that invisible portions 3 of the message 1 are deleted from the message 1 before message 1 is processed by spam filter 23. This ensures that spam filter 23 is operating on just visible portions 2 of the message 1, as is the human user 5. Thus, the above described technique by which spammers attempt to trick spam filters is foiled.

An example of how parser 21 performs steps 34 through 38 for an e-mail message 1 will now be described. In this example, the e-mail message 1 comprises:

<P><font color="white" background="black">PURCHASE <font background="white">CONFIRMATION FOR</font> VIAGRA</font></P>

Parser 21 sees the expression "<P>". This indicates the beginning of a new paragraph in HTML. There is no color information within this tag (it is not inline style), so parser 21 goes on to examine the next characters. The parser then sees "<font" (step 34). This tells parser 21 that a new color tag has been encountered. Parser 21 decodes the tag to mean that there is a white foreground on a black background. In one embodiment, parser 21 puts the expression "WhiteOnBlack" onto stack 25. Stack 25 may be a FILO (First In Last Out) stack. Parser 21, by means of semantic analysis, determines (step 36) that this combination is visible, and in one embodiment sets flag 26 to "visible". Since flag 26 is set to “visible”, parser 21 at step 38 sends the next word ("PURCHASE") to filter 23, either immediately or after the entire expression has been decoded. In the case where the next word is sent to filter 23 after the

entire expression has been decoded, parser 21 temporarily stores the next word in a buffer memory.

Next, parser 21 encounters (step 34) another color tag, indicating that the background color has changed to white. So now parser 21 knows through simple analysis (step 36) that the foreground and background colors are both white, and that the foreground is therefore invisible to the user 5. In one embodiment, parser 21 pushes the expression "WhiteOnWhite" onto the stack 25 and sets flag 26 to "invisible". Since flag 26 is set to “invisible”, parser 21 deletes (step 37) all characters until the next color tag, i.e., the characters "CONFIRMATION FOR", from message 1. Parser 21 then encounters an end-tag ("</font>"). This causes parser 21 to take the most recent item ("WhiteOnWhite") off stack 25. Now the item at the top of stack 25 is "WhiteOnBlack", so parser 21 resets flag 26 to "visible". Thus, parser 21 sends the next word (“VIAGRA”) to filter 23 at step 38.

The words "PURCHASE VIAGRA" are visible to the human user 5 since they comprise white text or black background, and the words "CONFIRMATION FOR" are invisible 3 to the human user 5, because they comprise white text on a white background. The spammer is attempting to feed the words "PURCHASE CONFIRMATION FOR VIAGRA" to spam filter 23, because many spam filters, upon seeing the words "PURCHASE CONFIRMATION", will treat the message 1 as being clean, thinking that user 5 has made a previous on-line purchase and that message 1 is simply a confirmation thereof. However, as can be seen from the above, the present invention has deleted the words "CONFIRMATION FOR" from message 1, and has sent just the words "PURCHASE VIAGRA" to the spam filter 23.

The above is a relatively simple example, wherein parser 21 can simply compare the words "white" and "black" to see whether the foreground and background colors are the same or substantially the same. When more sophisticated colors are used, color comparison module 22 is invoked to make this decision.

Instead of simple "white" and "black", the HTML can specify:

color="#001767"

This is hexadecimal notation for a dark purple. The numbers following the "#" comprise three components, each having two digits. All of these components can range from zero decimal to 255 decimal. The first two digits (00) specify the red component of the color, the second two digits (17) specify the green component of the color, and the last two digits (67) specify the blue component of the color. In decimal notation, this is equivalent to a red component of zero, a green component of 23, and a blue component of 103.

Similarly, the HTML can specify:

background="#0E147A"

This is also hexadecimal notation for a purple color wherein, in decimal notation, the red component is 14, the green component is 20, and the blue component is 122.

In one embodiment of the present invention, the red, green, and blue components are converted to hue, saturation, and brightness components using a conventional algorithm. This algorithmic conversion can be performed by parser 21 or by color comparison module 22. In the above example, red zero, green 23, blue 103 converts to a hue of 227 degrees, a saturation of 100%, and a brightness of 40%. Similarly, red 14, green 20, blue 122 converts to a hue of 237 degrees, a saturation of 89%, and a brightness of 48%. Color comparison module 22 is then invoked by parser 21, to determine whether the difference between the foreground color and the background color is negligible or not. The negligibility threshold can be pre-selected by trial and error, i.e., difference between foreground color and background color being "negligible" means that a typical human user 5 finds the foreground characters to be invisible.

In one embodiment, color comparison module 22 makes a distinction between gray-scale color and hued color. In this embodiment, gray-scale color comparison parameters are invoked whenever the saturation value of either the foreground or the background is zero, or when the saturation of both the foreground and the background is zero; and hued color comparison parameters are invoked in all other cases.

For gray-scale color, hue makes no difference. Only the saturation and brightness values need be compared. In one embodiment in which gray-scale color comparison parameters are invoked, if the difference in saturation values between the foreground and background is less than 5% and the difference in brightness values between the foreground and background is less than 4%, the foreground color is deemed to be invisible, i.e., the difference between the foreground color and background color is deemed to be negligible. These parameters are appropriate for when the display (monitor) associated with recipient user 5 is a CRT (Cathode Ray Tube). A CRT is weaker than an LCD (Liquid Crystal Display) monitor for gray-scale colors. For LCD monitors, appropriate criteria for declaring the foreground color to be invisible are that the saturation difference is less than 3% and the brightness difference is less than 2%.

For comparison of hued color values, in one embodiment, particularly useful when the recipient user's monitor is a LCD monitor, the foreground color is deemed to be invisible when the difference in hue between the foreground and background is less than 6 degrees, and the combined brightness and saturation difference is less than 14%. For hued colors, an LCD monitor is weaker than a CRT monitor, so, for a CRT monitor, in one embodiment, the foreground color is deemed to be invisible when the difference in hue between the foreground and background is less than 4 degrees, and the combined brightness and saturation difference between the foreground and background is less than 12%.

The above description is included to illustrate the operation of the preferred embodiments and is not meant to limit the scope of the invention. The scope of the invention is to be limited only by the following claims. From the above discussion, many variations will be apparent to one skilled in the art that would yet be encompassed by the spirit and scope of the present invention.

What is claimed is: